Arrests Data Cleaning

In [1]:

```
# Import Libraries
import pandas as pd
import numpy as np
pd.set_option('display.max_columns', None)
```

In [2]:

```
# Load Data
df = pd.read_csv("data/arrests_adult-arrests-details_arrestdetail.csv")
```

The data was obtained from City of Phoenix Open Data at: https://www.phoenixopendata.com/dataset/arrests

In [3]:

```
# Print first 5 rows of data
df.head()
```

Out[3]:

	ARST_NUM	DATE_OCCUR	DAY_OF_WEEK	MONTH	QTR	YEAR	ARREST_TYPE	UNIQUE_NAME_ID	SUBJ_SEX	SUBJ_RACE	SUBJ_ETHI
0	PHX201801013548	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	O	MNI-3059468	Male	White	H
1	PHX201801013538	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	T	MNI-100947779	Male	White	H
2	PHX201801013560	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	O	MNI-1270697	Male	White	Non-H
3	PHX201801013490	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	O	MNI-100947321	Male	White	H
4	PHX201801013488	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	T	MNI-100947309	Male	Asian / Pacific Islander	Non-H

In [4]:

```
# Print number of rows and columns
df.shape
```

Out[4]:

```
(238890, 35)
```

In [5]:

```
# Print data types
df.dtypes
```

Out[5]:

```
ARST_NUM                   object
DATE_OCCUR                 object
DAY_OF_WEEK                object
MONTH                      object
QTR                        object
YEAR                        int64
ARREST_TYPE                object
UNIQUE_NAME_ID             object
SUBJ_SEX                   object
SUBJ_RACE                  object
SUBJ_ETHNICITY             object
SIMPLE_SUBJ_RE_GRP         object
SUBJ_AGE                    int64
SUBJ_AGE_GROUP             object
ARST_OFFICER               object
ARST_OFFICER_SEX           object
ARST_OFFICER_RACE          object
SIMPLE_EMPL_RE_GRP         object
HUNDREDBLOCKADDR           object
PRECINCT_NUM              float64
PRECINCT                   object
BEAT_NUM                  float64
BEAT                       object
MAPGRID                    object
COUNCIL_DISTRICT_NUM      float64
COUNCIL_DISTRICT           object
FELONY_CHARGES              int64
MISDEMEANOR_CHARGES         int64
OTHER_CHARGES               int64
UNKNOWN_CHARGES             int64
P1VIOLENT_CHARGES           int64
P1PROPERTY_CHARGES          int64
P2DRUG_CHARGES              int64
ASSAULTOFFICER_CHARGES      int64
RESISTARST_CHARGES          int64
dtype: object
```

In [6]:

```
# Find number missing values for each column
df.isna().sum()
```

Out[6]:

```
ARST_NUM                     0
DATE_OCCUR                   0
DAY_OF_WEEK                  0
MONTH                        0
QTR                          0
YEAR                         0
ARREST_TYPE                  0
UNIQUE_NAME_ID               0
SUBJ_SEX                     0
SUBJ_RACE                    0
SUBJ_ETHNICITY               0
SIMPLE_SUBJ_RE_GRP           0
SUBJ_AGE                     0
SUBJ_AGE_GROUP               0
ARST_OFFICER                 2
ARST_OFFICER_SEX             5
ARST_OFFICER_RACE            5
SIMPLE_EMPL_RE_GRP           5
HUNDREDBLOCKADDR          3069
PRECINCT_NUM              6953
PRECINCT                  6953
BEAT_NUM                  7038
BEAT                      7038
MAPGRID                   7022
COUNCIL_DISTRICT_NUM      7105
COUNCIL_DISTRICT             0
FELONY_CHARGES               0
MISDEMEANOR_CHARGES          0
OTHER_CHARGES                0
UNKNOWN_CHARGES              0
P1VIOLENT_CHARGES            0
P1PROPERTY_CHARGES           0
P2DRUG_CHARGES               0
ASSAULTOFFICER_CHARGES       0
RESISTARST_CHARGES           0
dtype: int64
```

In [7]:

```
# Drop Council District Number since it is part of the name in the Council District column
df.drop(['COUNCIL_DISTRICT_NUM'], axis=1)
```

Out[7]:

	ARST_NUM	DATE_OCCUR	DAY_OF_WEEK	MONTH	QTR	YEAR	ARREST_TYPE	UNIQUE_NAME_ID	SUBJ_SEX	SUBJ_RACE	SUB
0	PHX201801013548	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	O	MNI-3059468	Male	White	
1	PHX201801013538	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	T	MNI-100947779	Male	White	
2	PHX201801013560	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	O	MNI-1270697	Male	White	
3	PHX201801013490	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	O	MNI-100947321	Male	White	
4	PHX201801013488	01/01/2018	2-MONDAY	01-JANUARY	Q1	2018	T	MNI-100947309	Male	Asian / Pacific Islander	
...	...	...	...	...	...	...	...	...	...	...	
238885	PHX202504308385	04/30/2025	4-WEDNESDAY	04-APRIL	Q2	2025	O	MNI-18139184	Male	American Indian / Alaskan Native	
238886	PHX202504308429	04/30/2025	4-WEDNESDAY	04-APRIL	Q2	2025	O	MNI-2924918	Male	Black	
238887	PHX202504308511	04/30/2025	4-WEDNESDAY	04-APRIL	Q2	2025	T	MNI-103151855	Male	Black	
238888	PHX202504308388	04/30/2025	4-WEDNESDAY	04-APRIL	Q2	2025	T	MNI-102696496	Male	White	
238889	PHX202504308396	04/30/2025	4-WEDNESDAY	04-APRIL	Q2	2025	O	MNI-102381219	Female	American Indian / Alaskan Native	

238890 rows × 34 columns

In [8]:

```
# Drop the numbers from Day of the Week and Month, leaving only the text
df['DAY_OF_WEEK'] = df['DAY_OF_WEEK'].str[2:]
df['MONTH'] = df['MONTH'].str[3:]
```

In [9]:

```
# CHeck all unique values for Month
df['MONTH'].unique()
```

Out[9]:
```
array(['JANUARY', 'FEBRUARY', 'MARCH', 'APRIL', 'MAY', 'JUNE', 'JULY',
       'AUGUST', 'SEPTEMBER', 'OCTOBER', 'NOVEMBER', 'DECEMBER'],
      dtype=object)
```

In [10]:
```python
# Check all unique values for Council District
df['COUNCIL_DISTRICT'].unique()
```

Out[10]:
```
array(['Council District 7', 'Council District 4', 'Council District 3',
       'Council District 5', 'Council District 6', 'Council District 1',
       'Council District 8', 'Council District 2', 'Council District NA'],
      dtype=object)
```

In [11]:
```python
# Convert the Council District NA to nulls
df.loc[df['COUNCIL_DISTRICT'].str.contains('Council District NA', na=False), 'COUNCIL_DISTRICT'] = np.nan
```

In [12]:
```python
# Check unique age groups
df['SUBJ_AGE_GROUP'].unique()
```

Out[12]:
```
array(['30s', '20s', '40s', '<20', '50s', '60s', '70s', '80s', '90s',
       '120s'], dtype=object)
```

In [13]:
```python
# Check all ages above 90
df[df['SUBJ_AGE'] > 90]
```

Out[13]:

	ARST_NUM	DATE_OCCUR	DAY_OF_WEEK	MONTH	QTR	YEAR	ARREST_TYPE	UNIQUE_NAME_ID	SUBJ_SEX	SUBJ_RACE	SU
227476	PHX202412027785	12/01/2024	SUNDAY	DECEMBER	Q4	2024	S	MNI-100952064	Male	White	
232961	PHX202502168162	02/16/2025	SUNDAY	FEBRUARY	Q1	2025	O	MNI-103152132	Male	White	

The 125 year old is probably a typo and is most likely a 25 year old, given the violence charge. We will leave the 92 year old person, since the values are possible even if unlikely

In [14]:
```python
# Example condition: ages above 100 are unrealistic
df.loc[df['SUBJ_AGE'] > 100, 'SUBJ_AGE'] = 25
df.loc[df['SUBJ_AGE_GROUP'].str.contains('120', na=False), 'SUBJ_AGE_GROUP'] = '20s'
```

In [15]:
```python
# Check that the row has been corrected
df[df['SUBJ_AGE'] > 90]
```

Out[15]:

	ARST_NUM	DATE_OCCUR	DAY_OF_WEEK	MONTH	QTR	YEAR	ARREST_TYPE	UNIQUE_NAME_ID	SUBJ_SEX	SUBJ_RACE	SU
227476	PHX202412027785	12/01/2024	SUNDAY	DECEMBER	Q4	2024	S	MNI-100952064	Male	White	

In [16]:

```
# Check that the row has been corrected
df.loc[df['ARST_NUM'] == 'PHX202502168162']
```

Out[16]:

	ARST_NUM	DATE_OCCUR	DAY_OF_WEEK	MONTH	QTR	YEAR	ARREST_TYPE	UNIQUE_NAME_ID	SUBJ_SEX	SUBJ_RACE	SUI
232961	PHX202502168162	02/16/2025	SUNDAY	FEBRUARY	Q1	2025	O	MNI-103152132	Male	White	

In [17]:

```
# Change the data types of Precinct and Beat Number
df['PRECINCT_NUM'] = pd.to_numeric(df['PRECINCT_NUM'], errors='coerce').astype('Int64')
df['BEAT_NUM'] = pd.to_numeric(df['BEAT_NUM'], errors='coerce').astype('Int64')
```

In [18]:

```
# Check the data types again
df.dtypes
```

Out[18]:

```
ARST_NUM                    object
DATE_OCCUR                  object
DAY_OF_WEEK                 object
MONTH                       object
QTR                         object
YEAR                         int64
ARREST_TYPE                 object
UNIQUE_NAME_ID              object
SUBJ_SEX                    object
SUBJ_RACE                   object
SUBJ_ETHNICITY              object
SIMPLE_SUBJ_RE_GRP          object
SUBJ_AGE                     int64
SUBJ_AGE_GROUP              object
ARST_OFFICER                object
ARST_OFFICER_SEX            object
ARST_OFFICER_RACE           object
SIMPLE_EMPL_RE_GRP          object
HUNDREDBLOCKADDR            object
PRECINCT_NUM                 Int64
PRECINCT                    object
BEAT_NUM                     Int64
BEAT                        object
MAPGRID                     object
COUNCIL_DISTRICT_NUM       float64
COUNCIL_DISTRICT            object
FELONY_CHARGES               int64
MISDEMEANOR_CHARGES          int64
OTHER_CHARGES                int64
UNKNOWN_CHARGES              int64
P1VIOLENT_CHARGES            int64
P1PROPERTY_CHARGES           int64
P2DRUG_CHARGES               int64
ASSAULTOFFICER_CHARGES       int64
RESISTARST_CHARGES           int64
dtype: object
```

In [19]:

```
# Print the first 5 rows of the data
df.head()
```

Out[19]:

	ARST_NUM	DATE_OCCUR	DAY_OF_WEEK	MONTH	QTR	YEAR	ARREST_TYPE	UNIQUE_NAME_ID	SUBJ_SEX	SUBJ_RACE	SUBJ_ETHI
0	PHX201801013548	01/01/2018	MONDAY	JANUARY	Q1	2018	O	MNI-3059468	Male	White	H
1	PHX201801013538	01/01/2018	MONDAY	JANUARY	Q1	2018	T	MNI-100947779	Male	White	H
2	PHX201801013560	01/01/2018	MONDAY	JANUARY	Q1	2018	O	MNI-1270697	Male	White	Non-H
3	PHX201801013490	01/01/2018	MONDAY	JANUARY	Q1	2018	O	MNI-100947321	Male	White	H
4	PHX201801013488	01/01/2018	MONDAY	JANUARY	Q1	2018	T	MNI-100947309	Male	Asian / Pacific Islander	Non-H

In [20]:

```
# Save the dataframe as a csv file for Tableau Visualization
df.to_csv('cleaned_arrest_data.csv', index=False, encoding='utf-8')
```